Precision Measurement Central (5)

Part 5: RS485 networking in practice

Design by J. Wickenhäuser www.wickenhaeuser.com

In the previous instalments of this series featuring the MSC1201 board and microcontroller, we have already discussed many aspects of using it for various applications, ranging from a development platform (for which it is ideally suited) to a precision sensor in a network. In this instalment, we examine how the elements discussed earlier can be combined to form a complete system or network.

You can use the software described here as the starting point for developing your own applications. Despite the use of print( ) statements and floating-point computations, the code written for the application described in this instalment occupies only around 5 kB (by comparison, the free evaluation version of the suggested uC/51 compiler generates up to 8 kB of code).

Figure 1 depicts a network consisting of two MSC1210 boards and an RS232/RS485 converter. This network is intended to work with cable lengths of up to 1,000 metres, with the two MSC nodes at one end and the RS232/RS485 converter at the other end. The network is also powered via the converter.

Construction aspects

Assuming a supply voltage of 15 V, each device connected to the network requires approximately 30 mA. In order for the nodes to function properly, the supply voltage must be at least 7.5 V. A certain amount of margin must be provided, since a

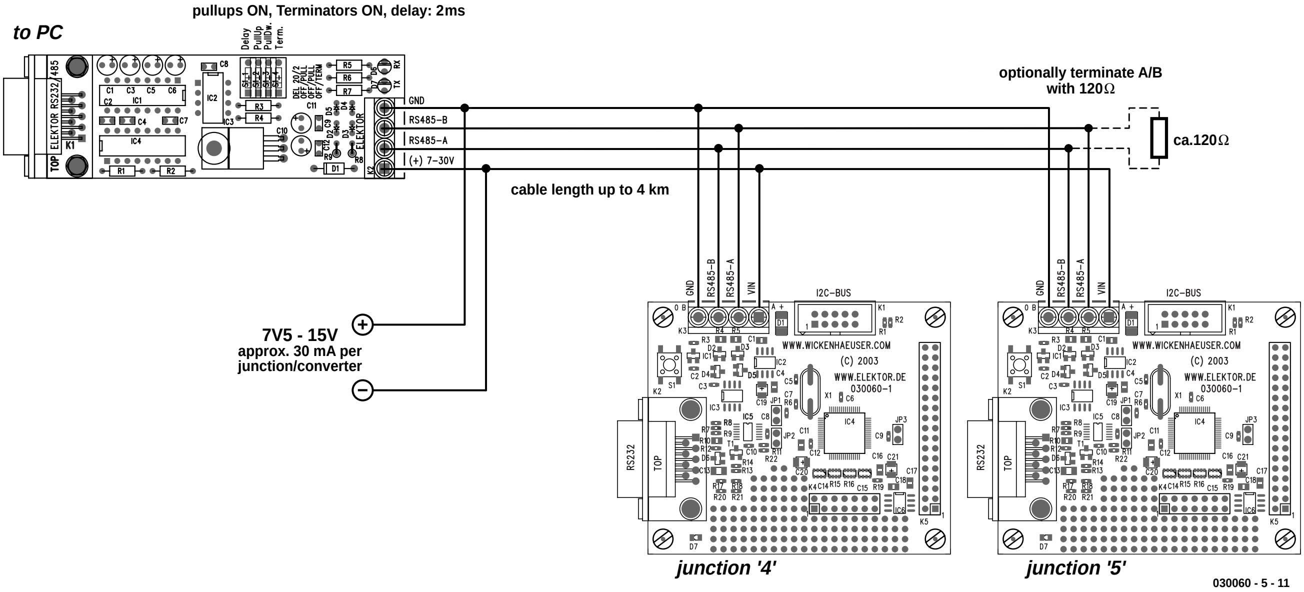
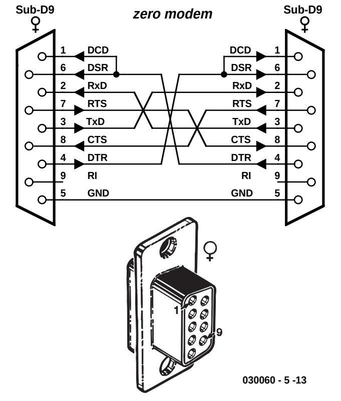


Figure 1. A network composed of two MSC1210 boards and an RS232/RS485 converter.

Null (zero) modem cable

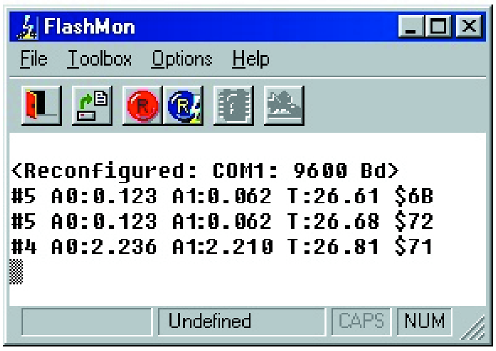


Almost all communications problems reported by readers in connection with the serial interface (RS232/V.24) can be traced back to the use of improper sub-D connectors and/or incorrect cables.

For the majority of microcontroller boards and projects with serial interfaces that have been described in *Elektor Electronics*, the RS232 connection is made using a 9-way sub-D socket (female connector) on the circuit board. In such cases the connecting cable is a simple serial interface cable (extension cable) with a socket at one end and a plug at the other end, with all pins connected one-to-one. This is also true of the connection between the PC and the RS485 adapter described here.

The MCS1210 forms an exception to this rule, since it is fitted with a sub-D plug (male connector). This means that a 'null modem' (or zero modem') cable must be used to connect the MSC1210 board to a PC. Such a cable has a 9-way socket (female connector) at each end. In addition, the connections between two sets of pins (1 & 4 and 2 & 3) are crossed over. Furthermore, pins 1 and 6 of each connector are directly connected to each other at the connector. The figure shows the wiring diagram of a null modem cable and the pin numbering of the 9-way sub-D socket (as seen from the front of the socket).

In order to be completely sure that you have the proper cable, you should check the connections using an ohmmeter or continuity tester. The jumpers between pins 1 and 6 must without fail be present in the connectors!



```
<Reconfigured: COM1: 9600 Bd>
#5 A0:0.123 A1:0.062 T:26.61 $6B
#5 A0:0.123 A1:0.062 T:26.68 $72
#4 A0:2.236 A1:2.210 T:26.81 $71
```

Figure 2. Measured values
A0: value at A0 input relative to AGND, in volts
A1: value at A1 input relative to AGND, in volts
T: chip temperature in °C

node may sometimes draw additional current, such as when transmitting. With a capacity of 100 mA on the bus, you will in any case be on the safe side.

With a total round-trip distance of 2000 metres, this means the voltage drop along the cable must not exceed 7.5 V. This yields a lead resistance (copper resistance) of approximately 75 Ω (assuming 100 mA). Since the specific resistance of copper is approximately 0.02 Ω·m/mm_, a four-conductor cable with a cross-sectional area (c.s.a.) of 0.6 mm_ is adequate. Fortunately, a cable with exactly these specifications is commercially available as a standard telecommunications cable (with a screen, no less!) at a relatively low cost.

In the case of very long cables, it may not be possible to power the entire RS485 bus from a single point. This was already discussed in the previous instalment. Thus before deciding to use a cable with 'super fat' conductors, you might be better advised to use a supplementary power supply.

With such long cables, it is normally necessary to terminate both ends of the RS485 bus. With the previously mentioned telecommunications cable, the suggested value of 120 Ω will almost always be correct.

Network software

Once the network has been physically implemented, you still need some software. RS485 requires a reliable data transmission protocol using the parameters 9600 (bit rate), 8, N, 1 (by contrast, the PC and the converter communicate via RS232 at 57,000 bit/s).

A hash mark (#) serves as the start character. This is followed by the address of the destination node for the command. After this come the actual data (in the form of readable text), followed at the end by a checksum that allows the recipient to determine whether the received data are correct. In our case, the checksum is marked by a $ character and consists of two bytes containing the lateral sum of all preceding characters after the start character. In normal practice, more complicated schemes are used, such as polynomial

A professional application

The company GeoPrecision manufactures sensor networks, among other products. The illustrated sensor has been specifically developed for use in underground mining. Due to its high accuracy and stability, an MSC microcontroller is used for this purpose.

The complete circuitry is housed inside a 52-mm stainless-steel enclosure. The enclosure is waterproof and hermetically sealed. Nevertheless, new software can be downloaded whenever desired, and the calibration factors are externally accessible. This capability is provided by a small operating system in each sensor, which manages the free space in the Flash memory. This means that up to 30 kB of program memory are available for individually programming each sensor. Distance is no longer an issue, thanks to the use of RS485.

Naturally, for such an application is it highly important for the transmitted data to be absolutely correct. Consequently, checksums are used intensively. Nevertheless, each sensor essentially contains a 'shrunken' version of the MSC board, for which all of the software has been developed using the uC/51 compiler.

List of articles

- Part 1: Precision Measurement Central (1): with an MSC1210 development platform (July/August 2003)
  Introduction to the overall concept of the MSC1210 experimenter's board
- Part 2: Precision Measurement Central (2): printed circuit board and software suite (September 2003)
  Printed circuit board, layout and components; software descriptions
- LC Display with I²C Bus (September 2003)
  I²C bus description and a user-friendly LC display for the MSC1210
- Part 3: Precision Measurement Central (3): Flash for all purposes (October 2003)
  The Flash memory and how to use it
- Part 4: Precision Measurement Central (4): RS485, or networking the MSC1210 micro (November 2003)
  Network architecture and a simple RS485/RS232 converter for the PC
- Part 5: Precision Measurement Central (5): RS485 networking in practice

checksums. There is a large amount of technical literature available on this subject in convenient 'pre-digested' form.

As already mentioned in the fourth installment, a brief pause must be inserted between activating the transmitter section and sending the start character, in order to avoid transmission errors. The PC converter uses a time window for activating the transmitter section. For this reason, the pull-up and pull-down resistors should be connected, since otherwise the first character might be transmitted incorrectly. The PC converter will thus be set to transmit when the first byte from the PC appears. All jumpers must be fitted in the PC converter, although the terminator is naturally only necessary if the converter is located at the end of the cable.

Figure 2 shows a typical data packet. This was generated by the 'perfectly ordinary' terminal emulator program FlashMon, which comes with the uC/51 compiler.

In order to test the network using a standard terminal emulator, we can go even a step further and ignore the checksum when transmitting. It will then only be contained in the response. The demo program can only handle one command, which displays the measured value — but at least the source code is free.

When multiple boards are used as network nodes, the software accompanying this article should downloaded to each of the MSC1210 boards, with each node being assigned its own ID (designated `node id` in the source code, with a default value of 5).

Now you can send a command to node 5 by typing *#5<CR>* (where <CR> refers to the Return key). The LED of this board will go on for approximately one second, while the LEDs of the other nodes will blink briefly. After this you will receive a measured value as shown in Figure 2. The node also outputs this information in a somewhat more extensive form via its (local) RS232 interface (which is used to download the software).

A new version of the demo software produced using version V1.10.12 of the compiler is now available in directory
...\scr\msc1210\Elmet\Elmet485\ELM_FLASH.

However, you do not necessarily have to download the entire compiler (12 MB) in one go, since the individual parts can also be downloaded from the *Elektor Electronics* website.

The MSC microcontrollers have many additional features, all of which are extremely interesting. With the elements that have been described in this series of articles, which now comes to a conclusion, you should not find it difficult to use them for your own purposes.

(030060-5)

Links:

www.wickenhaeuser.com
ANSI C compiler for the 8051 family, including the source code for all of the MSC1210 demo programs

www.elektor-electronics.co.uk
Source of fully assembled and tested MSC1210 boards

www.geoprecision.com
Several practical applications